CLAIMS:

1. A lattice comparison method comprising:

receiving first and second lattices of labels to be compared, each lattice defining alternative label sequences that represent a sequential signal and each lattice comprising a plurality nodes each associated with one or more labels and representing a point in the sequential signal at which the associated label occurs; and

comparing the first lattice with the second lattice by propagating a plurality of paths, each path representing a comparison between labels in the first lattice and labels in the second lattice, and each path having an associated accumulative value representing the closeness of the comparison;

wherein during the path propagation, said comparing step defines, for each node in the first lattice, a plurality of associated storage areas, each storage area associated with a first lattice node also being associated with a respective node in the second lattice and being operable to store, during the path propagation, an accumulative value representing the closeness of the comparison between labels in the first lattice up to the associated first lattice node and labels in the second lattice up to the associated second lattice node; and

wherein said comparing step uses said storage areas during the propagation of said paths.

2. A method according to claim 1, wherein each

lattice comprises an acyclic directed graph representing different label sequences that represent said sequential signal.

3. A method according to claim 1 or 2, wherein said comparing step propagates said paths by processing the nodes within the said first lattice in sequential order.

4. A method according to any preceding claim, wherein when propagating a path from a source node in said first lattice to a destination node in said first lattice, said comparing step updates and propagates accumulative values stored in the storage areas associated with the source node to at least the storage areas associated with the destination node.

5. A method according to claim 4, wherein during the propagation of said accumulative values, said comparing step compares the appropriate accumulative value in the storage area associated with the destination node with the updated accumulative value from the storage area associated with the source node.

6. A method according to claim 5, wherein said comparing step replaces the accumulative value in the storage area associated with the destination node with the updated accumulative value from the storage area associated with the source node if the updated

accumulative value is better than the accumulative value stored in the storage area associated with the destination node.

7. A method according to any of claims 4 to 6, wherein said comparing step updates the accumulative values stored in the storage areas associated with the source node to take into account for substitution of the corresponding labels in the first lattice and the corresponding labels in the second lattice.

8. A method according to any of claims 4 to 7, wherein said comparing step updates the accumulative values stored in the storage areas associated with the source node to take into account the insertion of labels in the first lattice and/or in the second lattice.

9. A method according to any of claims 4 to 8, wherein said comparing step updates the accumulative value stored in the storage areas associated with the source node to take into account the deletion of labels from the first lattice and/or from the second lattice.

10. A method according to any of claims 4 to 9, wherein said comparing step updates the accumulative value stored in the storage areas associated with the source node to take into account the substitution,

insertion and deletion of labels from the first lattice and/or second lattice and wherein the storage area to which an updated score is propagated depends upon whether a label is substituted, inserted or deleted from the first lattice and/or the second lattice.

11. A method according to any of claims 4 to 10, wherein said comparing step updates the accumulative values stored in the storage areas associated with the source node by comparing the corresponding labels in the first lattice with the corresponding labels in the second lattice.

12. A method according to claim 11, wherein said comparing step updates said accumulative values by using predetermined confusion data which defines measures of confusability between the different labels.

13. A method according to any of claims 4 to 12, wherein said first lattice is generated by a recognition unit and includes confidence data associated with said labels indicative of the confidence that said recognition unit correctly recognised the label and wherein said comparing step updates said accumulative values stored in the storage areas associated with the source node using the confidence data for the corresponding labels.

14. A method according to claim 13, wherein both said first and second lattices include said confidence data and wherein said comparing step updates said accumulative values using the confidence data for the respective labels from the first and second lattices.

15. A method according to any preceding claim, wherein said first and second signals are representative of time sequential signals.

16. A method according to claim 15, wherein said nodes within the said first and second lattices represent the start and/or end time of a label within the lattice.

17. A method according to claim 15 or 16, wherein said lattices are representative of speech and wherein said labels are representative of sub-word units.

18. A method according to claim 17, wherein said sub-word units comprise phonemes.

19. A method according to any preceding claim, further comprising the step of processing the accumulative values stored for a node, to determine a similarity measure representing the similarity between the first and second lattices.

20. A method according to claim 19, wherein the second lattice represents a longer sequence than the first lattice and wherein the processing step processes the accumulative values stored for the node to determine if the second lattice includes one or more portions similar to the first lattice.

21. A method according to claim 20, wherein said processing step compares the accumulative values in the storage areas of the node to identify values better than a predetermined threshold, to identify said one or more portions in the second lattice which are similar to the first lattice.

22. A method according to claim 21, wherein said processing step identifies said one or more portions by identifying the storage areas having an accumulative value better than said threshold.

23. A method according to claim 21 or 22, wherein the sequence length of the first lattice is known, wherein when said processing step identifies an accumulative value better than said threshold, the second lattice node associated with the identified accumulative value represents the end of said portion corresponding to said first lattice and wherein said processing step estimates a beginning of the portion within the second lattice using the known sequence length of the first lattice.

24. A method according to any of claims 19 to 23, wherein said processing step processes the accumulative values associated with an end node of the first lattice.

25. A method according to any preceding claim, wherein said comparing step performs a dynamic programming alignment and comparison between the first and second lattices.

26. A method according to any preceding claim, wherein the storage areas associated with a node in the first lattice are stored in a node table associated with the first lattice node.

27. A method according to claim 26, wherein said storage areas in said node tables are arranged in a sequential order defined by the sequential order of the associated nodes.

28. A method of searching a database comprising a plurality of information entries to identify information to be retrieved therefrom, each of said plurality of information entries having an associated annotation lattice, the method comprising:

receiving a query lattice representing an input query;

comparing the query lattice with each annotation

lattice using the method of any preceding claim to provide a set of comparison results; and

identifying said information to be retrieved from said database using the set of comparison results.

29. A method according to claim 28, wherein said identifying step identifies the information to be retrieved from said database by identifying the annotation lattice most similar to the query lattice.

30. A method according to claim 28 or 29, wherein said identifying step identifies the N most relevant information entries by identifying those information entries having an annotation lattice most similar to the query lattice.

31. A lattice comparison apparatus comprising:

means for receiving first and second lattices of labels to be compared, each lattice defining alternative label sequences that represent a sequential signal and each lattice comprising a plurality nodes each associated with one or more labels and representing a point in the sequential signal at which the associated label occurs; and

means for comparing the first lattice with the second lattice by propagating a plurality of paths, each path representing a comparison between labels in the first lattice and labels in the second lattice, and each path having an associated accumulative value

representing the closeness of the comparison;

wherein during the path propagation, said comparing means is operable to define, for each node in the first lattice, a plurality of associated storage areas, each storage area associated with a first lattice node also being associated with a respective node in the second lattice and being operable to store, during the path propagation, an accumulative value representing the closeness of the comparison between labels in the first lattice up to the associated first lattice node and labels in the second lattice up to the associated second lattice node; and

wherein said comparing means is operable to use said storage areas during the propagation of said paths.

32. A computer readable medium storing computer executable instructions for causing a programmable computer device to carry out the method of any of claims 1 to 30.

33. Computer executable instructions for causing a programmable computer device to carry out the method of any of claims 1 to 30.